

Decoding of Differential AG Codes

Kwankyu Lee

Abstract

The interpolation-based decoding that was developed for general evaluation AG codes is shown to be equally applicable to general differential AG codes. A performance analysis of the decoding algorithm, which is parallel to that of its companion algorithm, is reported. In particular, the decoding capacities of evaluation AG codes and differential AG codes are seen to be nicely interrelated. As an interesting special case, a decoding algorithm for classical Goppa codes is presented.

Index Terms

Algebraic geometry code, decoding algorithm, interpolation, Gröbner base.

I. INTRODUCTION

Let X be a smooth geometrically irreducible projective curve defined over a finite field \mathbb{F} of genus g . Let $\mathbb{F}(X)$ and Ω_X denote the function field and the module of differentials of X respectively. Let P_1, P_2, \dots, P_n be distinct rational points on X , and $D = P_1 + P_2 + \dots + P_n$. Let G be an arbitrary divisor on X , whose support is disjoint from that of D . Recall that $\mathcal{L}(G) = \{f \in \mathbb{F}(X) \mid (f) + G \geq 0\}$ and $\Omega(G) = \{\omega \in \Omega_X \mid (\omega) \geq G\}$. Then Goppa's famous two codes [1] are defined by

$$C_{\mathcal{L}}(D, G) = \{(f(P_1), f(P_2), \dots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

and

$$C_{\Omega}(D, G) = \{(\text{res}_{P_1}(\omega), \text{res}_{P_2}(\omega), \dots, \text{res}_{P_n}(\omega)) \mid \omega \in \Omega(-D + G)\},$$

which are respectively called evaluation AG code and differential AG code. It is well-known that $C_{\Omega}(D, G) = C_{\mathcal{L}}(D, G)^{\perp}$, whose proof [2] requires the Riemann-Roch theorem.

Theorem 1 (Riemann-Roch). *Let A be a divisor on X . Then*

$$\dim \mathcal{L}(A) = \deg(A) + 1 - g + \dim \Omega(A).$$

Moreover if $\deg(A) \geq 2g - 1$, then $\dim \Omega(A) = 0$, and if $\deg(A) < 0$, then $\dim \mathcal{L}(A) = 0$.

Though the two kinds of AG codes were equally created, their historical development has been somewhat unbalanced. The target of intensive researches on the bounds on the minimum distance and decoding algorithms up to the bounds were usually the differential code or rather the dual code to the evaluation code. Thus all known decoding algorithms for differential AG codes decode $C_{\mathcal{L}}(D, G)^{\perp}$ rather than $C_{\Omega}(D, G)$ itself in the sense that they work with the data called syndromes defined using functions in $\mathcal{L}(G)$. See [3], [4], [5], [6], [7] and many references therein. The bias is also reflected on the terms like primal AG code and dual AG code, which mean evaluation AG code and differential AG code respectively. In this respect, the recent result [8] on the unique decoding algorithm for evaluation AG codes is against to the trend and implies that the duality is not essential for decoding and for bounding the minimum distance of $C_{\mathcal{L}}(D, G)$. This makes one conceive of a decoding algorithm for $C_{\Omega}(D, G)$ that likewise does not rely on the duality, syndromes, or the space $\mathcal{L}(G)$.

In this paper, we present a fast unique decoding algorithm for general differential AG codes, which does not rely on the duality and does not use syndromes defined by the functions in $\mathcal{L}(G)$. Essentially it is the interpolation-based decoding algorithm for $C_{\mathcal{L}}(D, G)$ of [8] rewritten for the differential AG code $C_{\Omega}(D, G)$ based on the same principle ideas but using data derived from the relevant space of differentials. Specifically, the ring R defined in this paper (1) is the same R defined in [8], but instead of R -module $\bar{R} \subset \mathbb{F}(X)$, here we define another R -module $\bar{W} \subset \Omega_X$. Thus the decoding algorithm for differential AG codes works with the polynomials in $R \oplus \bar{W}$ while the algorithm for evaluation AG codes works in $R \oplus \bar{R}$. These changes aside, the basic principles underlying both decoding algorithms are exactly the same. Thus we achieve an equal and symmetric treatment of Goppa's two codes in decoding and bounding the minimum distance.

In Section II, we set up the algebraic framework in which the decoding algorithm works. In Section III, we present the decoding algorithm after a brief explanation of its structure. In Section IV, we report a performance analysis. It will be clearly seen that the framework and the algorithm itself resemble the corresponding ones in [8] so closely that, to avoid repetition, we do not prove that the algorithm works correctly nor provide the proofs for the assertions about performance analysis, but instead refer the reader to [8] for almost verbatim details of missing proofs. In Section IV-C, we will see that decoding capacities

K. Lee is with the Department of Mathematics and Education, Chosun University, Gwangju 501-759, Korea (e-mail: kwankyu@chosun.ac.kr). He was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2013R1A1A2009714).

of both decoding algorithms are nicely interrelated. In Section V, we give an explicit example. In Section VI, we consider decoding of Goppa codes. Recall that Goppa codes are subfield subcodes of differential AG codes on the projective line. Hence as a special case, we obtain a decoding algorithm for classical Goppa codes. This algorithm is interesting because Goppa codes are the main workhorse in the McEliece code-based cryptosystem and the speed of decryption is largely dependent on the efficiency of its decoding algorithm.

II. PRELIMINARIES

We assume the existence of a rational point Q distinct from the points in the support of D . Let

$$R = \bigcup_{s=0}^{\infty} \mathcal{L}(sQ) \subset \mathbb{F}(X). \quad (1)$$

For $f \in R$, let $\rho(f) = -v_Q(f)$. The Weierstrass semigroup at Q is then

$$\Lambda = \{\rho(f) \mid f \in R\} = \{\lambda_0, \lambda_1, \lambda_2, \dots\} \subset \mathbb{Z}_{\geq 0},$$

which is a numerical semigroup whose number of gaps, the positive integers not in Λ , is the genus g of X . Let γ be the smallest positive nongap, and let $\rho(x) = \gamma$ for some $x \in R$. For each $0 \leq i < \gamma$, let a_i be the smallest nongap such that $a_i \equiv i \pmod{\gamma}$ and $\rho(y_i) = a_i$ for some $y_i \in R$. By the properties of $\rho : R \rightarrow \mathbb{Z}_{\geq 0}$ inherited from the valuation v_Q of $\mathbb{F}(X)$, the set $\{y_0, y_1, \dots, y_{\gamma-1}\}$ forms a basis of R as a free module of rank γ over $\mathbb{F}[x]$, which we call the Apéry system of R . Hence $\{x^k y_i \mid k \geq 0, 0 \leq i < \gamma\}$ is a vector space basis of R over \mathbb{F} , whose elements are called the monomials of R . The monomials of R are in one-to-one correspondence with the nongaps in Λ . For $\lambda \in \Lambda$, we denote by $\varphi_\lambda \in R$ the unique monomial with $\rho(\varphi_\lambda) = \lambda$.

Notice that the ring R and the numerical semigroup Λ are the same as defined in [8]. Now come the definitions which are new but correspond to \bar{R} and $\bar{\Lambda}$ in [8]. Let

$$\bar{W} = \bigcup_{s=-\infty}^{\infty} \Omega(-D + G - sQ) \subset \Omega_X,$$

which is clearly a module over R . For a differential $\omega \in \bar{W}$, let $\delta(\omega)$ denote the smallest integer s such that $\omega \in \Omega(-D + G - sQ)$. Thus $\delta(\omega)$ is simply $v_Q(G) - v_Q(\omega)$. Let $\bar{\Omega} = \delta(\bar{W})$. Note that $\Lambda + \bar{\Omega} = \bar{\Omega}$, and in this sense $\bar{\Omega}$ is a numerical Λ -module. The integers in $\bar{\Omega}$ will also be called nongaps. As $\bar{\Omega}$ contains all large enough integers, for each $0 \leq i < \gamma$, there exists the smallest nongap b_i of $\bar{\Omega}$ such that $b_i \equiv i \pmod{\gamma}$ and $\delta(\bar{\omega}_i) = b_i$ for some $\bar{\omega}_i \in \bar{W}$. Using the valuative properties of δ , we also see that $\{\bar{\omega}_i \mid 0 \leq i < \gamma\}$ forms a basis of \bar{W} as a free module of rank γ over $\mathbb{F}[x]$. For $s \in \bar{\Omega}$, define $\bar{\varphi}_s = x^k \bar{\omega}_i$ for $i = s \bmod \gamma$ and $k = (s - b_i)/\gamma \geq 0$. Then $\delta(\bar{\varphi}_s) = s$. Thus $\{\bar{\varphi}_s \mid s \in \bar{\Omega}\} = \{x^k \bar{\omega}_i \mid k \geq 0, 0 \leq i < \gamma\}$ is a basis of \bar{W} over \mathbb{F} , and will be called the monomials of \bar{W} . The set $\{\bar{\omega}_i \mid 0 \leq i < \gamma\}$ is called the Apéry system of \bar{W} . One may observe that \bar{W} , as W , and $\bar{\Omega}$ were defined in [8] but not used for decoding.

Now let us consider the R -module

$$Rz \oplus \bar{W} = \{fz + \omega \mid f \in R, \omega \in \bar{W}\},$$

with indeterminate z . It is also a free $\mathbb{F}[x]$ -module of rank 2γ with free basis $K = \{y_i z, \bar{\omega}_i \mid 0 \leq i < \gamma\}$, and that every element of $Rz \oplus \bar{W}$ can be written as a unique \mathbb{F} -linear combination of the monomials in $\mathcal{M} = \{x^k y_i z, x^k \bar{\omega}_i \mid k \geq 0, 0 \leq i < \gamma\}$. So we can regard the elements of $Rz \oplus \bar{W}$ as polynomials over \mathbb{F} . We denote $\deg_x(x^k y_i z) = k$, $\deg_y(x^k y_i z) = i$, $\deg_x(x^k \bar{\omega}_i) = k$, and $\deg_{\bar{\omega}}(x^k \bar{\omega}_i) = i$.

Let us review the Gröbner basis theory on the R -module $Rz \oplus \bar{W}$. For an integer s , the weighted degree of a polynomial $fz + \omega \in Rz \oplus \bar{W}$ is defined by

$$\delta_s(fz + \omega) = \max\{\rho(f) + s, \delta(\omega)\}.$$

In particular, we have $\delta_s(x^k y_i z) = \gamma k + a_i + s$, $\delta_s(x^k \bar{\omega}_i) = \delta(x^k \bar{\omega}_i) = \gamma k + b_i$. Then δ_s induces the weighted degree order $>_s$ on \mathcal{M} , where we break ties by declaring the monomial with z precedes the other without z . For $f \in Rz \oplus \bar{W}$, the notations $\text{lt}_s(f)$, $\text{lm}_s(f)$, and $\text{lc}_s(f)$ are used to denote respectively the leading term, the leading monomial, and the leading coefficient, with respect to $>_s$. Suppose M is an $\mathbb{F}[x]$ -submodule of $Rz \oplus \bar{W}$. A finite subset B of M is called a *Gröbner basis* of M with respect to $>_s$ if the leading term of every element of M is an $\mathbb{F}[x]$ -multiple of the leading term of some element of B . We will write

$$B = \{g_i, f_i \mid 0 \leq i < \gamma\},$$

where we understand that the leading term of g_i is in \bar{W} while that of f_i is in Rz . The sigma set $\Sigma_s = \Sigma_s(M)$ of M is the set of all leading monomials of the polynomials in M with respect to $>_s$. The delta set $\Delta_s = \Delta_s(M)$ of M is the complement of Σ_s in \mathcal{M} .

The *residue* map

$$\text{res} : \bar{W} \rightarrow \mathbb{F}^n, \quad \omega \mapsto (\text{res}_{P_1}(\omega), \text{res}_{P_2}(\omega), \dots, \text{res}_{P_n}(\omega))$$

is linear over \mathbb{F} . Thus the differential AG code $C = C_\Omega(D, G) = \text{res}(\Omega(-D + G))$ is a linear code of length n over \mathbb{F} . Note that $\{\bar{\varphi}_s \mid s \in \bar{\Omega}, s \leq 0\}$ is a basis of $\Omega(-D + G)$ as a vector space over \mathbb{F} . Let k be the dimension of C . Then there is a set $S = \{s_0, s_1, \dots, s_{k-1}\} \subset \{s \mid s \in \bar{\Omega}, s \leq 0\}$ such that $\{\text{res}(\bar{\varphi}_s) \mid s \in S\}$ is a basis of C . Note that the map res is surjective onto \mathbb{F}^n . Indeed we can show that $\text{res}(\Omega(-D + G - sQ)) = \mathbb{F}^n$ for $s > |G| = \deg(G)$ by the Riemann-Roch theorem. Let $h_i \in \bar{W}$ be the differential such that $\text{res}(h_i)$ is the i th element of the standard basis of \mathbb{F}^n . Let J be the kernel of res . Note that J is a submodule of \bar{W} over R , and also over $\mathbb{F}[x]$. Let $\{\eta_i \mid 0 \leq i < \gamma\}$ be a Gröbner basis of J over $\mathbb{F}[x]$ such that $\deg_{\bar{\omega}}(\text{lt}(\eta_i)) = i$. Then we have

$$\sum_{0 \leq i < \gamma} \deg_x(\text{lt}(\eta_i)) = |\Delta(J)| = \dim_{\mathbb{F}} \bar{W}/J = n,$$

which corresponds to Proposition 2 of [8] and can be proved in a similar way.

III. DECODING ALGORITHM

We assume a codeword is sent through a communication channel and a vector $v \in \mathbb{F}^n$ is received. Thus we suppose $v = c + e$ with a codeword c and the error vector e . Then $c = \text{res}(\mu)$ with a unique differential

$$\mu = \sum_{s \in S} m_s \bar{\varphi}_s \in \Omega(-D + G).$$

For all integer $0 < s \in \bar{\Omega}$, let $v^{(s)} = v$, $c^{(s)} = c$, and $\mu^{(s)} = \mu$. For $s \in S$, let

$$\begin{aligned} \mu^{(s-1)} &= \mu^{(s)} - m_s \bar{\varphi}_s, \\ c^{(s-1)} &= c^{(s)} - \text{res}(m_s \bar{\varphi}_s), \\ v^{(s-1)} &= v^{(s)} - \text{res}(m_s \bar{\varphi}_s), \end{aligned}$$

and for $s \leq 0$ but $s \notin S$, let $\mu^{(s-1)} = \mu^{(s)}$, $c^{(s-1)} = c^{(s)}$, $v^{(s-1)} = v^{(s)}$. Then we have $\mu^{(s)} \in \Omega(-D + G - sQ)$, $c^{(s)} = \text{res}(\mu^{(s)})$, and $v^{(s)} = c^{(s)} + e$ for all s .

The *interpolation module* I_v for a vector $v = (v_1, v_2, \dots, v_n)$ is now defined by

$$I_v = \{fz + \omega \in Rz \oplus \bar{W} \mid f(P_i)v_i + \text{res}_{P_i}(\omega) = 0, 1 \leq i \leq n\}, \quad (2)$$

which is a submodule of $Rz \oplus \bar{W}$ over R . Let $h_v = \sum_{i=1}^n v_i h_i$ so that $\text{res}(h_v) = v$. Then we see that $I_v = R(z - h_v) + J$. Hence the set

$$\{\eta_i, y_i(z - h_v) \mid 0 \leq i < \gamma\} \quad (3)$$

is a Gröbner basis of I_v with respect to $>_{\delta(h_v)}$.

The Fast Decoding Algorithm for differential AG codes, displayed in Figure 1, starts with the basis (3) and iterates the substeps Pairing, Voting, and Rebasing, computing a Gröbner basis of $I_{v^{(s-1)}}$ from that of $I_{v^{(s)}}$ while $m_s \in \mathbb{F}$ is computed by majority voting for $s \in S$. Let

$$\nu(s) = \frac{1}{\gamma} \sum_{0 \leq i < \gamma} \max\{\delta(\eta_{i'}) - \rho(y_i) - s, 0\} = |\Delta(J) \cap \Sigma(R\bar{\varphi}_s)|$$

for $s \in S$, and define $d_\Omega = \min\{\nu(s) \mid s \in S\}$. Then it can be shown that $m_s = m_s$ if $2\text{wt}(e) < \nu(s)$ for $s \in S$, and hence the algorithm succeeds in iteratively computing m_s for all $s \in S$ if $\text{wt}(e) \leq \tau = \lfloor (d_\Omega - 1)/2 \rfloor$. The proof is mostly identical with the corresponding one in [8], with some obvious change of notations. So we leave out the proof.

Note that the Fast Decoding Algorithm for differential AG codes is also enhanced with the speedup techniques introduced in the Section III.E of [8] for evaluation AG codes. A minor difference is due to the different inequality in the following theorem.

Theorem 2. Suppose $\mu \in \Omega(-D + G - sQ)$ and $v \in \mathbb{F}^n$. If for a nonzero $f \in I_v$,

$$\delta_s(f) + \text{wt}(v - \text{res}(\mu)) + 2g - 1 \leq |G|,$$

then $f(\mu) = 0$. Here $f(\mu)$ denotes the f with the variable z substituted with μ .

Proof: Let $\delta_s(f) = d$. Then $f(\mu) \in \Omega(-D + G - dQ)$. As $f \in I_v$, $\text{res}_{P_i}(f(\mu)) = 0$ for i with $v_i = \text{res}_{P_i}(\mu)$. It follows that

$$f(\mu) \in \Omega(- \sum_{v_i \neq \text{res}_{P_i}(\mu)} P_i + G - dQ).$$

Hence by the Riemann-Roch theorem, if $f(\mu)$ is nonzero, then we must have $-\text{wt}(v - \text{res}(\mu)) + |G| - d < 2g - 1$. \square

According to Theorem 2, a polynomial f in $I_{v^{(s)}}$ satisfying the condition $\delta_s(f) + \text{wt}(e) + 2g - 1 \leq |G|$ is called a *Q-polynomial* for $v^{(s)}$. In the Fast Decoding Algorithm, we actually use the condition $\delta_s(f) + \tau + 2g - 1 \leq |G|$ since if we assume $\text{wt}(e) \leq \tau$, then a polynomial f in $B^{(s)}$ satisfying the condition is a *Q-polynomial* for $v^{(s)}$. See the step Q. Since $|\Delta_s(I_{v^{(s)}}) \cap Rz| \leq \text{wt}(e)$ (see Corollary 7 in [8]), if we have $|\Delta_s(I_{v^{(s)}}) \cap Rz| > \tau$ during decoding, then we must have $\text{wt}(e) > \tau$, and the decoder may declare *Decoding Failure*. See the step F.

Fast Decoding Algorithm. Let $v \in \mathbb{F}^n$ be the received vector.

S1 Compute $h_v = \sum_{i=1}^n v_i h_i$. Let $N = \delta(h_v)$, and set

$$B^{(N)} = \{\eta_i, y_i(z - h_v) \mid 0 \leq i < \gamma\}.$$

Let $\mathbf{m}_s = 0$ for s with $N < s \in S$. If $N \leq 0$, then set $\mathbf{m}_s \in \mathbb{F}$ such that $h_v = \sum_{s \in S} \mathbf{m}_s \bar{\varphi}_s$, and go to the step S3.

S2 Repeat the following for s from N to s_0 . Let $B^{(s)} = \{g_i^{(s)}, f_i^{(s)} \mid 0 \leq i < \gamma\}$ be a Gröbner basis of $I_{v^{(s)}}$ with respect to $>_s$ where

$$\begin{aligned} g_i^{(s)} &= \sum_{0 \leq j < \gamma} c_{i,j} y_j z + \sum_{0 \leq j < \gamma} d_{i,j} \bar{\omega}_j \\ f_i^{(s)} &= \sum_{0 \leq j < \gamma} a_{i,j} y_j z + \sum_{0 \leq j < \gamma} b_{i,j} \bar{\omega}_j \end{aligned}$$

and let $\nu_i^{(s)} = \text{lc}(d_{i,i})$.

F If $\sum_{0 \leq i < \gamma} \deg(a_{i,i}) > \tau$, then declare Decoding Failure and stop.

Q If there is an i such that $\gamma \deg(a_{i,i}) + a_i + \tau + 2g - 1 \leq |G|$, then set $f_i^{(s)} = \varphi z + \psi$. Set $\mathbf{m}_t \in \mathbb{F}$ such that $-\varphi^{-1}\psi = \sum_{t \in S, t \leq s} \mathbf{m}_t \bar{\omega}_t$, and proceed to the step S3. If ψ is not divisible by φ , then declare Decoding Failure and stop.

M Pairing. For $0 \leq i < \gamma$, let $i' = (i+s) \bmod \gamma$, $k_i = \deg(a_{i,i}) + (a_i + s - b_{i'})/\gamma$, and $c_i = \deg(d_{i',i'}) - k_i$.

Voting. If $s \notin S$, then for i with $k_i \geq 0$, let

$$m_i = -b_{i,i'}[x^{k_i}], \quad \mu_i = 1$$

and for i with $k_i < 0$, let $m_i = 0, \mu_i = 1$. Let $m = 0$ in both cases.

If $s \in S$, then for each i , let

$$m_i = -\frac{b_{i,i'}[x^{k_i}]}{\mu_i}, \quad \mu_i = \text{lc}(a_{i,i} y_i \bar{\varphi}_s)$$

and let $\bar{c}_i = \max\{c_i, 0\}$, and let m be the element of \mathbb{F} with the largest $\sum_{m=m_i} \bar{c}_i$, and let $\mathbf{m}_s = m$.

Rebasing. For each i , do the following. If $m_i = m$, then let

$$\begin{aligned} g_{i'}^{(s-1)} &= g_{i'}^{(s)}(z + m \bar{\varphi}_s) \\ f_i^{(s-1)} &= f_i^{(s)}(z + m \bar{\varphi}_s) \end{aligned}$$

and let $\nu_{i'}^{(s-1)} = \nu_{i'}^{(s)}$. If $m_i \neq m$ and $c_i > 0$, then let

$$\begin{aligned} g_{i'}^{(s-1)} &= f_i^{(s)}(z + m \bar{\varphi}_s) \\ f_i^{(s-1)} &= x^{c_i} f_i^{(s)}(z + m \bar{\varphi}_s) - \frac{\mu_i(m - m_i)}{\nu_{i'}^{(s)}} g_{i'}^{(s)}(z + m \bar{\varphi}_s) \end{aligned}$$

and let $\nu_{i'}^{(s-1)} = \mu_i(m - m_i)$. If $m_i \neq m$ and $c_i \leq 0$, then let

$$\begin{aligned} g_{i'}^{(s-1)} &= g_{i'}^{(s)}(z + m \bar{\varphi}_s) \\ f_i^{(s-1)} &= f_i^{(s)}(z + m \bar{\varphi}_s) - \frac{\mu_i(m - m_i)}{\nu_{i'}^{(s)}} x^{-c_i} g_{i'}^{(s)}(z + m \bar{\varphi}_s) \end{aligned}$$

and let $\nu_{i'}^{(s-1)} = \nu_{i'}^{(s)}$. Let $B^{(s-1)} = \{g_i^{(s-1)}, f_i^{(s-1)} \mid 0 \leq i < \gamma\}$.

S3 Output the codeword $\sum_{s \in S} \mathbf{m}_s \text{res}(\bar{\varphi}_s)$.

Fig. 1. Fast Decoding Algorithm for Differential AG Codes

IV. PERFORMANCE ANALYSIS

A. Decoding Capacity

Proposition 3. *We have $\nu(s) = |\Delta(J) \cup \Delta(R\bar{\varphi}_s)| - n + |G| - 2g + 2 - s$. Hence d_Ω is at least Goppa bound $|G| - 2g + 2$ for differential AG codes.*

Proof: Note that

$$\nu(s) = |\Delta(J) \cap \Sigma(R\bar{\varphi}_s)| = |\Delta(J) \cup \Delta(R\bar{\varphi}_s)| - |\Delta(R\bar{\varphi}_s)| = |\Delta(J) \cup \Delta(R\bar{\varphi}_s)| - n + |G| - s - 2g + 2.$$

To see the last equality, note that by the Riemann-Roch theorem,

$$|\Delta(R\bar{\varphi}_s)| = \dim_{\mathbb{F}} \overline{W}/R\bar{\varphi}_s = \dim_{\mathbb{F}} \Omega(-D + G - (t+s)Q)/\mathcal{L}(tQ)\bar{\varphi}_s = n - |G| + s + 2g - 2$$

for all large enough t . Since $|\Delta(J) \cup \Delta(R\bar{\varphi}_s)| \geq n$, we have $\nu(s) \geq |G| - 2g + 2 - s$. Now our assertion follows. \square

Let us define $\tau_M(s) = \lfloor (\nu(s) - 1)/2 \rfloor$ for $s \in \Omega$, which is the largest number of errors for which the majority voting succeeds for s . Like Proposition 22 in [8], we can show that for nongap $s \leq |G| - 2g + 2$,

$$\lfloor (|G| - 2g + 1 - s)/2 \rfloor \leq \tau_M(s) \leq \lfloor (|G| - g + 1 - s)/2 \rfloor \quad (4)$$

and if $s \leq |G| - 4g + 2$, equality holds on the left.

We now find out when the condition in the step Q is satisfied. Let $t = \text{wt}(e)$.

Theorem 4. *Let $B^{(s)}$ be a Gröbner basis of $I_{v^{(s)}}$ with respect to $>_s$. If $\lambda_t + s + t + 2g - 1 \leq |G|$, then there exists an $f \in B^{(s)}$ such that f is a Q -polynomial for $v^{(s)}$.*

Proof: By the same argument of the proof of Lemma 18 of [8], we know that there exists an $f \in B^{(s)}$ such that $\delta_s(f) \leq \lambda_t + s$. So if $\lambda_t + s + t + 2g - 1 \leq |G|$, then this f satisfies the condition to be a Q -polynomial. \square

Let us define

$$\tau_Q(s) = \max\{t \mid \lambda_t + s + t + 2g - 1 \leq |G|\}$$

for $s \in \bar{\Omega}$. By Theorem 4, the value $\tau_Q(s)$ is the largest number of errors for which a Q -polynomial exists in $B^{(s)}$. Like Proposition 20 in [8], we can then show that for $s \in \bar{\Omega}$,

$$\lfloor (|G| - 3g + 1 - s)/2 \rfloor \leq \tau_Q(s) \leq \lfloor (|G| - 2g + 1 - s)/2 \rfloor \quad (5)$$

and for $s \leq |G| - 5g + 1$, equality holds on the left. From (4) and (5), we see that $\tau_Q(s) \leq \tau_M(s)$ for nongap $s \leq |G| - 2g + 2$, and moreover for $s \leq |G| - 5g + 1$,

$$\tau_M(s) - \tau_Q(s) = \begin{cases} \lceil g/2 \rceil & \text{if } |G| - s \text{ is odd,} \\ \lfloor g/2 \rfloor & \text{if } |G| - s \text{ is even.} \end{cases}$$

Recall that the actual condition used in the step Q to find a Q -polynomial in $B^{(s)}$ is $\delta_s(f) + \tau + 2g - 1 \leq |G|$. By a similar proof of Theorem 4, we have

Theorem 5. *Suppose $t = \text{wt}(e) \leq \tau$. If $\lambda_t + s + \tau + 2g - 1 \leq |G|$, then there exists an $f \in B^{(s)}$ satisfying $\delta_s(f) + \tau + 2g - 1 \leq |G|$, and f is a Q -polynomial for $v^{(s)}$.*

By Theorem 5, the condition in the step Q is satisfied for some $s \geq s_Q(t) = |G| - \lambda_t - \tau - 2g + 1$ depending on $t = \text{wt}(e)$, and at the latest for some $s \geq s_Q(\tau) = |G| - \lambda_\tau - \tau - 2g + 1$. Finally note that

$$\tau = \min_{s \in S} \tau_M(s), \quad s_Q(\tau) = \max\{s \mid \tau_Q(s) \geq \tau\}, \quad (6)$$

which can be verified by definitions.

B. Complexity

The Fast Decoding Algorithm iteratively updates a $2\gamma \times 2\gamma$ array of polynomials in $\mathbb{F}[x]$ that represents $B^{(s)}$. Each of the 2γ rows of the array are again viewed as pairs of vectors in $\mathbb{F}[x]^\gamma$. For the initialization step S1, we precompute h_i for $1 \leq i \leq n$ and η_i for $0 \leq i < \gamma$ in the vector form. In the *Rebasing* substep of the step M, the most intensive computation is the substitution of z with $z + m\bar{\varphi}_s$. As $\bar{\varphi}_s$ is in the form $x^k \bar{\omega}_i$, the computation is facilitated if $y_i \bar{\omega}_j$ for $0 \leq i, j < \gamma$ is precomputed in the vector form. For the step S3, it is necessary to precompute the vectors $\text{res}(\bar{\varphi}_{s_i})$ in \mathbb{F}^n for $0 \leq i \leq k - 1$, essentially the generator matrix of the code C . Our complexity analysis is now summarized, omitting the details, in the following.

Proposition 6. (1) *Lagrange basis polynomial h_i can be chosen such that the maximum degree of the polynomials in the vector form of h_i is bounded by $N_h = \lfloor (n + 2g - 1)/\gamma \rfloor$.*

(2) *The maximum degree of the polynomials in the vector form of η_i is bounded by $N_\eta = \lfloor (n + 3g + \gamma - 1)/\gamma \rfloor$.*

(3) The maximum degree of the polynomials in the $2\gamma \times 2\gamma$ array during an execution is bounded by $N_{\deg} = 1 + \lfloor (n + 4g - 2)/\gamma \rfloor$ if $g > 0$. If $g = 0$, then it is bounded by n .

(4) The number of iterations is at most $N_{\text{iter}} = n + 2g$.

Proposition 7. If $g > 0$, an execution of the Fast Decoding Algorithm takes $O((n + 4g + \gamma)(2\tau + 3g)(2g + \gamma)\gamma)$ multiplications. For $g = 0$, it takes $O(n^2)$ multiplications. The implicit constant is absolute.

Observe that these results are exactly the same with the complexity analysis of the decoding algorithm for evaluation AG codes reported in [8].

C. Comparisons of minimum distance bounds

We now show that d_Ω is indeed a lower bound for the minimum distance of the code $C_\Omega(D, G)$. Recall that $d_\Omega = \min\{\nu(s) \mid s \in S\}$ where $\nu(s) = |\Delta(J) \cap \Sigma(R\bar{\varphi}_s)| = n - |\Delta(J) \cap \Delta(R\bar{\varphi}_s)|$.

Theorem 8. The minimum distance d of $C_\Omega(D, G)$ is lower-bounded by d_Ω .

Proof: Let $s \in S$ and suppose $c = \text{res}(\mu)$, $\mu = \sum_{t \in S, t \leq s} a_t \bar{\varphi}_t \in \Omega(-D + G)$ with nonzero $a_s \in \mathbb{F}$. Consider \mathbb{F}^n as an \mathbb{F} -algebra with the component-wise multiplication $*$. Let us consider the evaluation map from R to \mathbb{F}^n , which is a surjective homomorphism of \mathbb{F} -algebras. Let \tilde{J} be the kernel of the map, and we have an isomorphism R/\tilde{J} with \mathbb{F}^n . Then

$$\begin{aligned} n - \text{wt}(c) &= \dim_{\mathbb{F}}\{v \in \mathbb{F}^n \mid v * c = 0\} \\ &= \dim_{\mathbb{F}}\{f \in R/\tilde{J} \mid \text{res}(f\mu) = 0\} \\ &= \dim_{\mathbb{F}}\{f \in R/\tilde{J} \mid f\mu \in J\} \\ &= \dim_{\mathbb{F}} \ker(R/\tilde{J} \xrightarrow{\mu} \bar{W}/J) \\ &= n - \dim_{\mathbb{F}} \text{im}(R/\tilde{J} \xrightarrow{\mu} \bar{W}/J) \\ &= n - \dim_{\mathbb{F}}(J + R\mu)/J \\ &= \dim_{\mathbb{F}} \bar{W}/(J + R\mu) \\ &= |\Delta(J + R\mu)| \end{aligned}$$

Hence $\text{wt}(c) = n - |\Delta(J + R\mu)|$. Now as $\Sigma(R\mu) = \Sigma(R\bar{\varphi}_s)$, we have $|\Delta(J + R\mu)| \leq |\Delta(J) \cap \Delta(R\bar{\varphi}_s)|$. Therefore $\text{wt}(c) \geq n - |\Delta(J) \cap \Delta(R\bar{\varphi}_s)| = \nu(s) \geq d_\Omega$. This result implies that $d \geq d_\Omega$. \square

Recall that

$$\begin{aligned} \Lambda &= \{i \mid \mathcal{L}(iQ) \neq \mathcal{L}(i-1)Q\}, \\ \bar{\Lambda} &= \{s \mid \mathcal{L}(G + sQ) \neq \mathcal{L}(G + (s-1)Q)\}, \\ \bar{\Omega} &= \{s \mid \Omega(-D + G - sQ) \neq \Omega(-D + G - (s-1)Q)\}, \end{aligned}$$

where $\bar{\Lambda}$ is the numerical Λ -module used in [8]. Then Theorem 33 in [8] says that the minimum distance of $C_{\mathcal{L}}(D, G)$ is lower bounded by

$$d_{\mathcal{L}} = \min_{s \in \bar{\Lambda}, s \leq 0} |\{(i, j) \mid i \in \Lambda, j \in \bar{\Omega}, i + j + s = 1\}|.$$

For comparison of d_Ω with $d_{\mathcal{L}}$, we assume $\deg(G) \geq 2g - 1$ such that $S = \{s \in \bar{\Lambda}, s \leq 0\}$ from now on. Like Lemma 32 of [8], we can show that $a + 1 \in \bar{\Lambda}$ if and only if $-a \notin \bar{\Omega}$ or $\bar{\varphi}_{-a} \in \Delta(J)$, which implies that

$$d_\Omega = \min_{s \in \bar{\Omega}, s \leq 0} |\{(i, j) \mid i \in \Lambda, j \in \bar{\Lambda}, i + j + s = 1\}|.$$

Observe the nice symmetry between $d_{\mathcal{L}}$ and d_Ω .

Recall that $\text{ev}(f) = (f(P_1), f(P_2), \dots, f(P_n))$ for $f \in \bar{R} = \bigcup_s \mathcal{L}(G + sQ)$. Let $C_i = \text{ev}(\mathcal{L}(G + iQ)) \subset \mathbb{F}^n$ for $i \in \mathbb{Z}$. The codes C_i are sometimes used in formulating minimum distance bounds of AG codes [9], [10].

Lemma 9. $s \in \bar{\Omega}$ if and only if $-s + 1 \notin \bar{\Lambda}$ or $C_{-s} \neq C_{-s+1}$.

Proof: By the Riemann-Roch theorem,

$$\begin{aligned} s \in \bar{\Omega} &\iff \dim_{\mathbb{F}} \Omega(-D + G + (-s + 1)Q) \neq \dim_{\mathbb{F}} \Omega(-D + G - sQ) \\ &\iff \dim_{\mathbb{F}} \mathcal{L}(-D + G + (-s + 1)Q) = \dim_{\mathbb{F}} \mathcal{L}(-D + G - sQ) \\ &\iff -s + 1 \notin \bar{\Lambda} \text{ or } C_{-s+1} \neq C_{-s} \end{aligned}$$

For the last equivalence, recall that $\dim_{\mathbb{F}} C_i = \dim_{\mathbb{F}} \mathcal{L}(G + iQ) - \dim_{\mathbb{F}} \mathcal{L}(-D + G + iQ)$. \square

Lemma 9 allows us to write

$$d_\Omega = \min_{C_k \neq C_{k-1}, k \geq 1} |\{(i, j) \mid i \in \Lambda, j \in \bar{\Lambda}, i + j = k\}|,$$

$$d_{\mathcal{L}} = \min_{s \in \Lambda, s \leq 0} |\{i \in \Lambda \mid C_{i+s} \neq C_{i+s-1}\}|.$$

Finally let us focus on one-point AG codes with $G = mQ$. In this case $\bar{\Lambda} = \Lambda - m$. Let $H_i = \text{ev}(\mathcal{L}(iQ)) = C_{i-m}$ and $H^* = \{i \mid H_i \neq H_{i-1}\} \subset \Lambda$. Then

$$d_\Omega = \min_{k \in H^*, k > m} |\{(i, j) \mid i, j \in \Lambda, i + j = k\}| \geq \min_{k \in H^*, k > m} |\{(i, j) \mid i, j \in H^*, i + j = k\}|,$$

$$d_{\mathcal{L}} = \min_{s \in \Lambda, s \leq m} |\{i \in \Lambda \mid i + s \in H^*\}| \geq \min_{s \in \Lambda, s \leq m} |\{i \in H^* \mid i + s \in H^*\}|,$$

where the former is the famous Feng-Rao bound of the one-point AG code $C_\Omega(D, mQ)$ and the latter is a slight variation of the bound d^* defined in [9].

V. EXAMPLE

Let X be the Hermitian curve of genus $g = 3$ defined by the equation

$$y^3 + y = x^4$$

over $\mathbb{F}_9 = \mathbb{F}_3(\alpha)$ with $\alpha^2 - \alpha - 1 = 0$. Let $G = -O + 18Q$ where O is the origin and Q is the unique point at infinity. Except O and Q , there are 26 rational points

$$\begin{aligned} & (0, \alpha^2), (0, \alpha^6), (1, 2), (1, \alpha), (1, \alpha^3), (2, 2), (2, \alpha), \\ & (2, \alpha^3), (\alpha, 1), (\alpha, \alpha^7), (\alpha, \alpha^5), (\alpha^2, 2), (\alpha^2, \alpha), (\alpha^2, \alpha^3), \\ & (\alpha^7, 1), (\alpha^7, \alpha^7), (\alpha^7, \alpha^5), (\alpha^5, 1), (\alpha^5, \alpha^7), (\alpha^5, \alpha^5), \\ & (\alpha^3, 1), (\alpha^3, \alpha^7), (\alpha^3, \alpha^5), (\alpha^6, 2), (\alpha^6, \alpha), (\alpha^6, \alpha^3). \end{aligned}$$

Then the differential AG code $C = C_\Omega(D, G)$ is a $[26, 11, 13]$ linear code over \mathbb{F}_9 . Note that this is the dual code of $C_{\mathcal{L}}(D, G)$ dealt in Section IV-A of [8]. Hence the data about R are the same, but the data about \bar{W} are new.

The Weierstrass semigroup at Q is

$$\Lambda = \{0, 3, 4, 6, 7, 8, 9, \dots\}.$$

So $\gamma = 3$, and we take $x = x$. The Apéry system of R is

$$\begin{aligned} y_0 &= 1, & \rho(y_0) &= 0, \\ y_1 &= y, & \rho(y_1) &= 4, \\ y_2 &= y^2, & \rho(y_2) &= 8. \end{aligned}$$

On the other hand,

$$\bar{\Omega} = \{-13, -10, -9, -7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, \dots\},$$

and the Apéry system of \bar{W} is

$$\begin{aligned} \bar{\omega}_0 &= y(x^9 - x)^{-1} dx, & \delta(\bar{\omega}_0) &= -9, \\ \bar{\omega}_1 &= y^2(x^9 - x)^{-1} dx, & \delta(\bar{\omega}_1) &= -5, \\ \bar{\omega}_2 &= (x^9 - x)^{-1} dx, & \delta(\bar{\omega}_2) &= -13. \end{aligned}$$

The monomials of $Rz \oplus \bar{W}$ are displayed in the array below, where the common z factor of the monomials of Rz are omitted.

| | | | | | | | | | | | |
|-------|---|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| y_2 | 1 | x | x^2 | x^3 | x^4 | x^5 | x^6 | x^7 | x^8 | x^9 | \dots |
| y_1 | 1 | x | x^2 | x^3 | x^4 | x^5 | x^6 | x^7 | x^8 | x^9 | \dots |
| y_0 | 1 | x | x^2 | x^3 | x^4 | x^5 | x^6 | x^7 | x^8 | x^9 | \dots |

| | | | | | | | | | | | |
|------------------|---|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| $\bar{\omega}_2$ | 1 | x | x^2 | x^3 | x^4 | x^5 | x^6 | x^7 | x^8 | x^9 | \dots |
| $\bar{\omega}_1$ | 1 | x | x^2 | x^3 | x^4 | x^5 | x^6 | x^7 | x^8 | x^9 | \dots |
| $\bar{\omega}_0$ | 1 | x | x^2 | x^3 | x^4 | x^5 | x^6 | x^7 | x^8 | x^9 | \dots |

Here are the corresponding nongaps of Λ and \bar{W} , respectively.

| | | | | | | | | | | |
|---|----|----|----|----|----|----|----|----|----|---------|
| 8 | 11 | 14 | 17 | 20 | 23 | 26 | 29 | 32 | 35 | \dots |
| 4 | 7 | 10 | 13 | 16 | 19 | 22 | 25 | 28 | 31 | \dots |
| 0 | 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | \dots |

| | | | | | | | | | | |
|-----|-----|----|----|----|----|----|----|----|----|---------|
| -13 | -10 | -7 | -4 | -1 | 2 | 5 | 8 | 11 | 14 | \dots |
| -5 | -2 | 1 | 4 | 7 | 10 | 13 | 16 | 19 | 22 | \dots |
| -9 | -6 | -3 | 0 | 3 | 6 | 9 | 12 | 15 | 18 | \dots |

The Lagrange interpolation polynomials in \overline{W} are

$$\begin{aligned} h_1 &= (-x^8 + 1)\bar{\omega}_2 + (\alpha^2 x^8 + \alpha^6)\bar{\omega}_0, \\ h_2 &= (-x^8 + 1)\bar{\omega}_2 + (\alpha^6 x^8 + \alpha^2)\bar{\omega}_0, \\ &\vdots \\ h_{26} &= (\alpha^6 x^8 + \alpha^3 x^7 + \cdots + 1)\bar{\omega}_2 \\ &\quad + (\alpha^6 x^7 + 2x^6 + \cdots + 1)\bar{\omega}_1 \\ &\quad + (\alpha^3 x^8 + \alpha x^7 + \cdots + \alpha^5 x)\bar{\omega}_0. \end{aligned}$$

The $\mathbb{F}_9[x]$ -submodule J of \overline{W} has Gröbner basis $\{\eta_0, \eta_1, \eta_2\}$, where

$$\begin{aligned} \eta_0 &= (x^9 - x)\bar{\omega}_0, \\ \eta_1 &= (x^8 - 1)\bar{\omega}_1 + (x^8 - 1)\bar{\omega}_2, \\ \eta_2 &= (x^9 - x)\bar{\omega}_2. \end{aligned}$$

Using these data, we can compute

| s | $\nu(s)$ | s | $\nu(s)$ |
|-----|----------|-----|----------|
| 0 | 13 | -6 | 19 |
| -1 | 14 | -7 | 20 |
| -2 | 15 | -9 | 22 |
| -3 | 16 | -10 | 23 |
| -4 | 17 | -13 | 26 |
| -5 | 18 | | |

and hence $d_\Omega = 13$. The Fast Decoding Algorithm runs with the received vector v as input and with the above precomputed data. The decoding process itself is similar to that of the example in Section IV-A in [8]. Thus we omit an example run of the algorithm.

VI. DECODING GOPPA CODES

Let $L = \{\alpha_1, \alpha_2, \dots, \alpha_n\} \subset \mathbb{F}$ be a set of distinct rational points of the projective line $\mathbb{P}_{\mathbb{F}}$ over $\mathbb{F} = \mathbb{F}_{q^m}$, and let P_∞ denote the point at infinity. Let D denote the divisor of zeros of $\prod_{k=1}^n (x - \alpha_k)$. Let $g(x) \in \mathbb{F}[x]$ be a given polynomial with $g(\alpha_i) \neq 0$ for $1 \leq i \leq n$, and let Z be the divisor of zeros of $g(x)$. We assume $\deg g(x) < n$. Then the classical Goppa code $C|_{\mathbb{F}_q}$ is a subfield subcode over \mathbb{F}_q of the differential AG code

$$C = C_\Omega(D, Z - P_\infty).$$

As C is an $[n, k, d]$ code with $k = n - \deg g(x)$ and $d \geq \deg g(x) + 1$, the Goppa code $C|_{\mathbb{F}_q}$ is an $[n, k', d']$ linear code over \mathbb{F}_q with $k' \geq n - m \deg g(x)$, $d' \geq \deg g(x) + 1$. See [2] for more on Goppa codes and subfield subcodes of AG codes.

Clearly, the Fast Decoding Algorithm can decode the subfield subcode $C|_{\mathbb{F}_q}$ just by decoding C . So let us specialise the algorithm for C . Let $X = \mathbb{P}_{\mathbb{F}}$. The genus of the projective line is $g = 0$, and $\mathbb{F}(X) = \mathbb{F}(x)$, $R = \mathbb{F}[x]$, $\Lambda = \mathbb{Z}_{\geq 0}$. For $f(x) \in \mathbb{F}[x]$, simply $\rho(f(x)) = \deg f(x)$. In particular, $\rho(x) = \gamma = 1$ and $\rho(y_0) = a_0 = 0$ with $y_0 = 1$. Furthermore $\Omega_X = \mathbb{F}(x)dx$ and we can show that $\bar{W} = \mathbb{F}[x]\bar{\omega}_0$ and $\bar{\Omega} = \mathbb{Z}_{\geq 0} + b_0$ where

$$\bar{\omega}_0 = \frac{g(x)}{\prod_{i=1}^n (x - \alpha_i)} dx, \quad b_0 = \deg g(x) - n + 1,$$

noting that $\delta(f(x)\bar{\omega}_0) = \deg f(x) + b_0$. For $\omega = f(x)\bar{\omega}_0 \in \bar{W}$, we have

$$\text{res}(\omega) = (f(\alpha_1)g(\alpha_1), f(\alpha_2)g(\alpha_2), \dots, f(\alpha_n)g(\alpha_n))$$

Therefore $J = \mathbb{F}[x]\eta_0$ where

$$\eta_0 = \prod_{i=1}^n (x - \alpha_i)\bar{\omega}_0$$

and for $1 \leq i \leq n$,

$$h_i = \frac{\prod_{j=1, j \neq i}^n (x - \alpha_j)}{g(\alpha_i) \prod_{j=1, j \neq i}^n (\alpha_i - \alpha_j)} \bar{\omega}_0.$$

With the above precomputed data, the Fast Decoding Algorithm for Goppa codes is reduced to the following simple algorithm.

Fast Decoding Algorithm for Goppa Codes. Let $v \in \mathbb{F}^n$ be the received vector.

S1 Compute $h_v = \sum_{i=1}^n v_i h_i$. Let $G^{(n-1)} = \eta_0$, $F^{(n-1)} = z - h_v$ and $\nu^{(n-1)} = 1$.

S2 Repeat the following for s from $n-1$ to 0.

M Suppose $G^{(s)} = Cz + D\bar{\omega}_0$, $F^{(s)} = Az + B\bar{\omega}_0$. Let $k = \deg(A) + s$, and $c = \deg(D) - k$.

If $s \geq n - \deg g(x)$, then let $m = -B[x^k]$. If $m = 0$, let

$$G^{(s-1)} = G^{(s)} \quad F^{(s-1)} = F^{(s)},$$

and $\nu^{(s-1)} = \nu^{(s)}$. If $m \neq 0$ and $c > 0$, then let

$$G^{(s-1)} = F^{(s)} \quad F^{(s-1)} = \nu^{(s)} x^c F^{(s)} + m G^{(s)}$$

and let $\nu^{(s-1)} = -m$. If $m \neq 0$ and $c \leq 0$, then let

$$G^{(s-1)} = G^{(s)} \quad F^{(s-1)} = \nu^{(s)} F^{(s)} + m x^{-c} G^{(s)}$$

and let $\nu^{(s-1)} = \nu^{(s)}$.

If $s < n - \deg g(x)$, then let $\mathbf{m}_s = -\text{lc}(A)^{-1} B[x^k]$ and let

$$G^{(s-1)} = G^{(s)}(z + \mathbf{m}_s x^s \bar{\omega}_0) \quad F^{(s-1)} = F^{(s)}(z + \mathbf{m}_s x^s \bar{\omega}_0)$$

and let $\nu^{(s-1)} = \nu^{(s)}$.

S3 Let $\mu = \sum_{s=0}^{n-\deg g(x)-1} \mathbf{m}_s x^s$ and $r = (\mu(\alpha_i)g(\alpha_i) \mid 1 \leq i \leq n)$. If $r \in \mathbb{F}_q^n$, output r . Otherwise declare Decoding Failure.

Note that the differential $\bar{\omega}_0$, as well as z , is just a placeholder, and the algorithm actually works with a 2×2 array of univariate polynomials. We made some changes in notations for clarification and restructured the algorithm slightly for ready implementation. It should be noted that when the algorithm is in second phase $s < n - \deg g(x)$, the update of $G^{(s)}$ is actually unnecessary.

As is well-known, the binary Goppa code $C|_{\mathbb{F}_2}$ defined by a separable polynomial $g(x) \in \mathbb{F}_{2^m}(x)$ is identical with the Goppa code defined by $g(x)^2$. Hence $C|_{\mathbb{F}_2}$ has dimension $\geq n - m \deg g(x)$ and minimum distance $2 \deg g(x) + 1$. Moreover the Goppa code $C|_{\mathbb{F}_2}$ can be decoded up to $\deg g(x)$ number of errors by the Fast Decoding Algorithm for the Goppa code defined by $g(x)^2$. We take this approach in the following example.

Example 1. Let $C|_{\mathbb{F}_2}$ be the binary Goppa code defined by $L = \{0, 1, \alpha, \alpha^2, \dots, \alpha^6\} \subset \mathbb{F}_8 = \mathbb{F}_2[\alpha]$ with $\alpha^3 + \alpha + 1 = 0$ and the generator polynomial $g(x)^2 \in \mathbb{F}_8[x]$ with $g(x) = z^2 + z + 1$. This is an $[8, 2, 5]$ Goppa code, capable of correcting two errors. Precomputed are $\eta_0 = (x^8 + x)\bar{\omega}_0$ and

$$\begin{aligned} h_1 &= (x^7 + 1)\bar{\omega}_0, \\ h_2 &= (x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x)\bar{\omega}_0, \\ &\vdots \\ h_8 &= (\alpha x^7 + x^6 + \alpha^6 x^5 + \alpha^5 x^4 + \alpha^4 x^3 + \alpha^3 x^2 + \alpha^2 x)\bar{\omega}_0 \end{aligned}$$

Suppose $v = (1, 1, 1, 1, 1, 1, 0)$ is the received vector. The algorithm iteratively computes the following for $s = 7, 6, \dots, 0$.

$$\begin{aligned} G^{(7)} &= & 0z &+ & (x^8 + x)\omega_0 \\ F^{(7)} &= & 1z &+ & (\alpha x^7 + \alpha^2 x^5 + \alpha^4 x^4 + \alpha^5 x^3 + \alpha^3 x^2 + \alpha^2 x + 1)\omega_0 \\ G^{(6)} &= & 0z &+ & (\alpha x^7 + \alpha^2 x^5 + \alpha^4 x^4 + \alpha^5 x^3 + \alpha^3 x^2 + \alpha^2 x + 1)\omega_0 \\ F^{(6)} &= & xz &+ & (\alpha^2 x^6 + \alpha^4 x^5 + \alpha^5 x^4 + \alpha^3 x^3 + \alpha^2 x^2 + \alpha^3 x)\omega_0 \\ &\vdots & & & \\ G^{(0)} &= & xz &+ & (\alpha^2 x^6 + \alpha^4 x^5 + \alpha^4 x^4 + \alpha x^3 + \alpha^2 x^2 + \alpha^3 x)\omega_0 \\ F^{(0)} &= & (\alpha^3 x^2 + \alpha^5 x + \alpha^4)z &+ & (\alpha^3 x^2 + \alpha^5 x + \alpha^4)\omega_0 \end{aligned}$$

and in the second phase, it computes $\mathbf{m}_3 = 1, \mathbf{m}_2 = 1, \mathbf{m}_1 = 0, \mathbf{m}_0 = 1$. Then from $\mu = 1 + x^2 + x^3$, the algorithm computes the corrected codeword $(1, 1, 1, 1, 0, 1, 0, 0)$ to output.

VII. FINAL REMARKS

We presented a fast unique decoding algorithm for differential AG codes. The principle of the algorithm is the same with the companion algorithm for evaluation AG codes in [8], and the description of the algorithm is almost identical with that of

the other. In this paper, we focused on the differences and omitted most of the repetitive parts. The list decoding extension for evaluation AG codes done by [11] can be done for differential AG codes in the same way.

This work is partially done while I visited Maria Bras-Amorós. The author thanks her for helpful discussions.

REFERENCES

- [1] V. D. Goppa, “Codes on algebraic curves,” *Sov. Math. Dokl.*, vol. 24, no. 1, pp. 170–172, 1981.
- [2] H. Stichtenoth, *Algebraic Function Fields and Codes*, 2nd ed. Springer-Verlag, 2009.
- [3] G. L. Feng and T. T. N. Rao, “Decoding algebraic-geometric codes up to the designed minimum distance,” *IEEE Trans. Inf. Theory*, vol. 39, no. 1, pp. 37–45, 1993.
- [4] I. M. Duursma, “Majority coset decoding,” *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 1067–1070, 1993.
- [5] S. Sakata, H. E. Jensen, and T. Høholdt, “Generalized Berlekamp-Massey decoding of algebraic-geometric codes up to half the Feng-Rao bound,” *IEEE Trans. Inf. Theory*, vol. 41, no. 6, pp. 1762–1768, 1995.
- [6] T. Høholdt, J. H. van Lint, and R. Pellikaan, “Algebraic geometry of codes,” in *Handbook of coding theory, Vol. I, II*. Amsterdam: North-Holland, 1998, pp. 871–961.
- [7] P. Beelen and T. Høholdt, “The decoding of algebraic geometry codes,” in *Advances in algebraic geometry codes*, ser. Ser. Coding Theory Cryptol. World Sci. Publ., Hackensack, NJ, 2008, vol. 5, pp. 49–98.
- [8] K. Lee, M. Bras-Amorós, and M. E. O’Sullivan, “Unique decoding of general AG codes,” *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2038–2053, 2014.
- [9] O. Geil, C. Munuera, D. Ruano, and F. Torres, “On the order bounds for one-point AG codes,” *Adv. Math. Commun.*, vol. 5, no. 3, pp. 489–504, 2011.
- [10] O. Geil, R. Matsumoto, and D. Ruano, “Feng-Rao decoding of primary codes,” *Finite Fields Appl.*, vol. 23, pp. 35–52, 2013.
- [11] —, “List decoding algorithms based on Gröbner bases for general one-point AG codes,” in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, Jul. 2012, pp. 86–90.